International Journal of Computer Science and Engineering (IJCSE)
ISSN(P): 2278-9960; ISSN(E): 2278-9979
Vol. 3, Issue 1, Jan 2014, 73-84

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

CHALLENGES AND ATTACKS IN MP2P NETWORKS: A REVIEW

GULSHAN SIRKECK[1] & SURENDER SHARMA[2]

[1]Research Student, Department of Computer Science and Engineering, Shoolini University, Himachal Pradesh, India

[2]Associate Professor, Department of Computer Science and Engineering, Shoolini University, Himachal Pradesh, India

ABSTRACT

As we know Mobile Ad hoc networks (MANET) and Peer-to-Peer (P2P) networks share central characteristics such as their distributed and decentralized nature. Mobile Ad hoc Networks (MANETs) and Peer-to-Peer (P2P) networks share concepts of self-organization and decentralization. Both systems operate without a central, coordinating entity and do not require a preexisting communication infrastructure for operation. Combining both networking paradigms results in a Mobile Peer-to-Peer (MP2P) system that operates independently from a preexisting infrastructure. Securing MP2P networks in terms of availability and robustness as basic demands in envisioned application scenarios like first responder operations is a challenging task. In this paper, we present a review of architectures, challenges and attacks in these networks.

KEYWORDS: Cross-Layered Architectures, DHT Peer-to-Peer Network, Mobile Ad-Hoc Networks, Mobile Peer-to-Peer Networks, Tapestry, Sybil Attacks

INTRODUCTION

Mobile Ad hoc Networks (MANETs) and Peer-to-Peer (P2P) networks share concepts of self-organization and decentralization. Both systems operate without a central, coordinating entity and do not require a preexisting communication infrastructure for operation. Considering these similarities, combining MANETs and P2P networks is reasonable to obtain a fully distributed, decentralized and infrastructure-less communication substrate. Several architectures for such mobile P2P (MP2P) networks already exist. The proposed approaches differ in terms of the layer(s) on which they are implemented and in terms of the deployed mechanisms for the lookup process that is responsible for locating services available in the system as well as in terms of the self-organized network maintenance. Combining both architectures at a single layer or enabling a cross-layer information exchange may increase network efficiency since the overhead for network operation and maintenance can be reduced. Mostly, an MP2P network benefits from its inherent self-x features as, e.g., a single node failure will (most probably) not affect the whole network. Yet, on the other hand, securing decentralized networks in common and MP2P networks in particular is a challenging task.

The routing algorithms for both P2P networks and MANETs have to rely on the benign behavior of the nodes forming the network to forward messages to the destination correctly. In both architectures, nodes join and leave the network in a flexible way. Due to this, both are susceptible to routing or Sybil attacks. On the other hand, various MANET and P2P related security measures were proposed to increase the robustness of the routing algorithms and to detect Sybil attacks. However, the question on which layer the count ermeasures has to be implemented to optimize the network security and efficiency in MP2P networks is still open. In this paper, we review different approaches proposed for establishing Mobile P2P (MP2P) networks and briefly summarize the basics of the underlying MANET and P2P architectures.

FOUNDATIONS

Peer-to-Peer

P2P networks are used for file sharing, voice chat and many more services nowadays. While we focus on Distributed Hash Tables (DHT) in this paper, several other architectures were also proposed.

We basically have to distinguish between *structured* and *unstructured* networks [5].

Distributed Hashtables (DHT) are structured P2P networks in which each node is assigned an ID (e.g., by hashing its IP address). By this ID, nodes are identified and can be looked-up within the P2P network. During lookup, cooperation of other nodes in the network is required, as in the routing tables that are used to route lookup requests only the addresses of a fraction of the available nodes in the network is stored. While the particular lookup algorithm differs depending on the specific architecture, most DHTs scale logarithmically to the network size regarding the routing table size and the number of hops required to forward a request from source to destination.

Unstructured P2P networks can be further subdivided into *centralized*, *pure*, and *hybrid P2P* networks.

Centralized P2P networks require a central entity for the lookup of the provided services (e.g., locating stored objects in a file sharing scenario). Requests for content and notifications of new content have to be sent to this entity which stores the IP addresses of nodes offering services, thus providing lookup information. This way, every request can be served after a single hop in the P2P overlay but at the cost of a very high load at the central entity and a single point of failure. Napster [4] is a well-known example of a centralized P2P network.

While centralized P2P systems still contain elements of traditional Client/Server architectures, all nodes in pure P2P networks are treated equally. Flooding-based lookup algorithms are required as the source node is unaware of the logical location of services available in the P2P overlay (i.e., the requesting peer has no knowledge on which peer offers the requested service). No single point of failure exists and no complex lookup algorithm is required. Yet, a considerable lookup overhead is introduced since the lookup requests are flooded through the network. Free net [1] and Gnutella 0.4 [1] are examples for pure P2P networks.

The third kind of unstructured architectures are *hybrid P2P networks*. In hybrid networks, we distinguish between leaf nodes and super peers. Every super peer is responsible for a set of leaf nodes and maintains information about services provided by these nodes. Lookup requests are usually not flooded in the whole network but sent to the super peers. This results in a reduced overhead at the cost of a more complex lookup algorithm compared to *pure P2P* networks. Gnutella 0.6 [1] and KaZaA [1] are examples for *hybrid P2P* networks.

MANETs

MANETs are self-configuring networks established by mobile wireless nodes. In order to transmit messages to nodes which are not in direct transmission range, nodes between source and destination have to operate as router and forward the messages. No preexisting infrastructure is required. Like P2P networks, MANETs work in a distributed and decentralized way, thus avoiding single points of failure as the routing tasks of each node may be performed by nearby nodes. On the other hand, MANETs require special routing algorithms and several challenges regarding security occur due to the characteristics of the network. These challenges have to be addressed to achieve a reliable availability of the network which is required in envisioned application scenarios where no infrastructure is available, like first responder scenarios (e.g., *DUMBO [5], HiMoNN [6]*), development projects (e.g. *One Laptop per Child*) or car-to-car communication (e.g., *CAR 2 CAR Communication Consortium*).

Mobile Peer-to-Peer

As MANETs and P2P networks share central characteristics, combing both leads to a fully distributed architecture. Those MP2P networks consist of a MANET underlay combined with a P2P overlay. The P2P algorithm provides a lookup service on the MANET underlay. The major challenges in developing MP2P architectures are the overhead caused by the routing table maintenance combined with the strongly limited resources of the MANET underlay.

ARCHITECTURES

Peer-to-Peer

Structured P2P networks provide a good trade-off between complexity and scalability. Many DHT algorithms were proposed which can be distinguished by the network topology. We will introduce the most common topologies in the following.

Ring

Ring based architectures as, e.g., Chord distribute the node-IDs on a circular ID-space as shown in Figure 1(a). In Chord, every object with an object-ID *o* is stored at the node with the next higher node-ID *n* ($n > o$). The lookup is based on the node-ID and is done by forwarding the request clockwise on the circular ID-space. To offer shortcuts to nodes located far away in the ring topology, Chord uses a finger table that stores the required mapping of IP addresses to distant node IDs (the fingers are represented by arrows in Figure 1(a)).

The finger table of node *n* is structured in *i* rows where each row provides the address of the node with the ID $nodeIDi = nodeIDn + 2^{i-1}$. This results in lookup algorithm that is scalable to the network size regarding the routing table size, as only a small part of the networks nodes are stored at the finger table, and to the hop length of the request due to the structure of the finger table [2].

Tree

Tapestry is a well-known example for a DHT that is based on a tree architecture. Each node maintains a prefix-based lookup table. Each row *i* of the routing table links to nodes with *i* matching prefixes as shown in Figure 1(b). E.g., Node 013 provides links to Nodes 002, 021 and 033 at the second row ($i = 1$) [2].

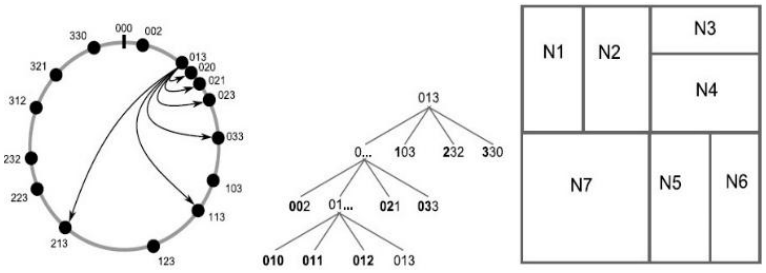
Whenever a node initiates or receives a lookup request, the object-ID is compared with the node-ID. The length of the matching prefix determines the row of the routing table where the address of the next hop node is stored.

This way, the size of the routing table as well as the efficiency of the lookup algorithm in terms of overlay hops required to deliver a lookup request scale logarithmically with respect to the network size in number of peers.

Hypercube

The Content Addressable Network (CAN) is a hypercube architecture that is based on a d-dimensional ID-space. Nodes are not identified by a singular point in this d-dimensional space but are represented by an area within the ID-space (as shown in Figure 1(c)) [2]. Objects, on the other hand, are assigned a singular point in the ID-space for which the node that is assigned to this area is responsible for.

The lookup algorithm of CAN is kept simple as every node forwards the request to the direct neighbor in the direction of the object's d-dimensional coordinates.



a) Ring (Chord) **b) Tree (Tapestry)** **c) Hypercube (CAN)**

Figure 1: DHT Architectures

Hybrid

A hybrid architecture combines the algorithms of other architectures. Pastry for example uses a tree-based primary routing table and a ring-based leaf set. While the primary routing table determines how to forward requests to nodes that are far away in terms of the difference of the node IDs, the leaf set is used when requests have to be forwarded in the direct neighborhood.

MANETs

Several routing algorithms for MANETs were proposed in the last years. These may be distinguished by their routing characteristics. For this paper, we differentiate between *proactive*, *reactive*, *hybrid*, and *geographical* algorithms.

In *proactive* approaches, routing tables are updated periodically and, thus, fresh routes to all nodes in the network are readily available (if the network is not partitioned). Due to this, no routing delay occurs before (and during) data transmission. On the other hand, overhead is generated as routing tables must be maintained regularly due to node mobility. The *proactive* Optimized Link State Routing Protocol (OLSR) [7], as an often referenced example, utilizes a link state algorithm for routing but also reduces overhead by selecting Multipoint Relays (MPR). Every node has to determine its MPRs by identifying the smallest set of direct neighbors required to cover every second hop neighbor as shown in Figure 2. In order to reduce the routing overhead, link state messages are sent to the MPRs only.

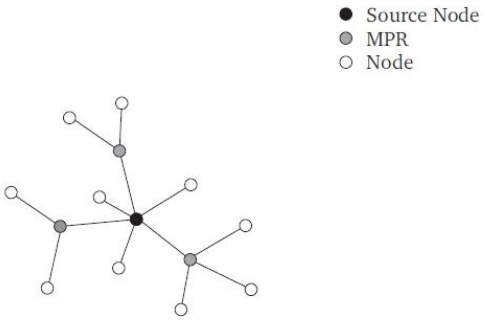


Figure 2: MPRS Select by an OLSR Node

Reactive algorithms, in contrast to *proactive* approaches, start routing for a specific destination only on demand, not unless a route is required for data transmission. As the topology in MANETs may change constantly due to node mobility, routing overhead may be significantly reduced when a route discovery is initiated only when a route is required. On the downside, when data has to be transmitted to a node for which no entry in the routing table exists, a route has to be established before the first packet can be sent. The Ad hoc On-Demand Distance Vector (AODV) [8] routing protocol is a well-known and often referenced *reactive* protocol for MANETs. Here, when a route is required, the source node has to broadcast a Route Request message (RREQ). Each node in transmission range has to check whether it is the destination of this request or whether a valid route to the destination is stored in the routing table. If so, the node sends a Route Reply

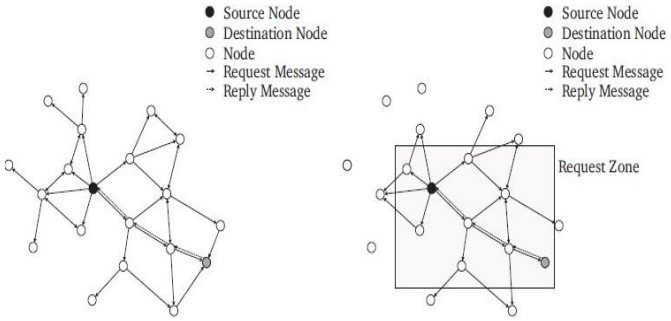
message (RREP) to the source. Otherwise the RREQ is re-broadcasted (as shown in Figure 3(a)). A node drops duplicates of RREQs to prevent loops. A route is established when the source node receives the RREP.

Hybrid protocols combine proactive and reactive approaches. While routing to nodes close to the source is done in a *proactive* way, routing to nodes that are further away is performed *reactively*. This way, routing overhead is reduced and a communication with nodes close to the source is not affected by routing delay. The Zone Routing Protocol (ZRP) is a well-known *hybrid* protocol. Routes to nodes within a specific local zone are maintained *proactively* while routes to nodes further away are established on demand.

When the destination is not within the local zone, a request is sent to the nodes at the border of this zone. Those nodes check their own local zone for the requested node and forward the request to their own border nodes again if the node could not be found within their zone. This proceeds until the destination node is found.

When the network nodes are aware of their *geographical* position, routing can be optimized by harnessing this information. The geographical position of the destination node can be used to limit the dissemination of routing messages to a specific area. This results in a reduced routing overhead as routing messages are not broadcasted in the whole network. For example, the Location-Aided Routing (LAR) determines an expected zone around the last known position of a destination node. Based on the expected zone, a request zone is defined around the position of the source node and the expected zone.

Route requests are broadcasted in this request zone only and, due to this, routing overhead is reduced (as shown in Figure 3(b)). An alternative approach, LAR proposes varying the request zone by intermediate nodes as they may have more recent information about the position of the destination node.



a) Reactive **b) Geographical**

Figure 3: MANET Routing Algorithm

Mobile Peer-to-Peer

While under lays based on (mostly) wired and infrastructure-based systems such as the Internet provide sufficient bandwidth for DHTs even when routing tables have to be maintained constantly due to the changing topology, resources of MANETs as underlay are more limited. Further, as the topology changes more often due to the mobility of nodes in MP2P networks, routing tables have to be maintained more frequently. Due to this, more control traffic is generated in an MP2P network compared to a traditional infrastructure-based P2P network but less bandwidth is available.

Therefore, MP2P architectures have to reduce the overhead generated by the routing algorithm and the routing table maintenance. Recently, several approaches were proposed to combine MANETs and P2P networks. The resulting MP2P architectures can be distinguished by the implementation of the P2P algorithm into *layered*, *integrated* and *cross-layered* approaches as shown in Figure 4

Layered

Layered architectures are a straightforward approach for adapting traditional P2P concepts to MP2P. Here, the MANET underlay and the P2P overlay are strictly separated; no cross-layer information is harnessed to optimize the system. As the routing and maintenance overhead of traditional P2P architectures would exceed the strongly limited resources of the MANET underlay, several techniques to reduce the maintenance overhead were proposed for *layered* approaches.

In order to reduce overhead on mobile nodes with limited resources, the Mobile Stealth DHT architecture proposed in distinguishes between mobile and static nodes. Static nodes are preferred for data storage and routing, while mobile nodes only have to manage the traffic generated by their own requests. This way, most overhead is shifted to the static nodes which are assumed to have stronger resources than mobile nodes. Further, the overhead generated by object relocation as a result of the mobility of nodes (mobility related churn) is avoided. This results in a reduced number of sent messages and an increased availability of objects. To influence the routing maintenance phase directly, Castro *et al*. propose the Bamboo DHT as P2P overlay of an MP2P network [10]. The Bamboo routing algorithm is similar to Pastry but differs in the routing maintenance. While Pastry updates the routing tables whenever the topology changes, Bamboo maintains the routing tables at specific time intervals. By increasing these intervals, routing related traffic is reduced.

Several approaches for MP2P networks harness location awareness of nodes to minimize routing overhead. Whenever node-IDs are generated based on the geographical position of the nodes, the routing algorithm may benefit from this information. The Peer Net [16] architecture splits the ID-space in several equal zones and assigns each virtual zone to a geographical area. Virtual residences are introduced as a geographical area where a node may be found with the highest probability. A proxy is required for each virtual residence to provide information about the nodes that reside in this virtual area. Each mobile node stores a mobility profile at the proxy of the virtual residence when leaving this zone. The mobility profile provides information based on which the current geographical position of the node can be estimated. As Peer Net is combined with the geographical MANET routing algorithm PILOT [15], overhead may be reduced as no flooding based routing algorithm is required. Another example for geographical routing is the location aware Bamboo DHT presented by Millar *et al*. This approach uses a land marking system to assign node-IDs to nodes. Land marking architectures split the ID-space in equal sized groups and assign a landmark-ID to a specific node within this group.

This node broadcasts the land marking signal. Each node within a land marking zone forwards the land marking signal. A node that receives a land marking message of a foreign landmark group compares the distance to the foreign landmark node with the distance to its own landmark node. If the new landmark node is closer, the node changes the node-ID and joins the closer landmark zone. As each landmark zone assigns node-IDs with a specific prefix, nodes with matching prefixes are physically close to each other. Due to this, a location aware routing is combined with a reduction of the overhead of the routing table maintenance phase by the Bamboo DHT. A major advantage of a location aware routing is the reduced overhead as the number of underlay hops is reduced.

Integrated

Integrated approaches reduce routing overhead by combining the MANET and P2P protocols at the network layer as shown in Figure 4(c). This way, the number of required routing tables is reduced as no separate MANET and P2P routing tables are required. Further, the routing algorithm may benefit from synergy effects such as updating overlay routing tables based on overheard underlay messages as described below.

By reducing the complexity and the number of required entries in the routing table, routing and maintenance

overhead may be reduced. Further, knowledge of the physical neighborhood may improve the routing efficiency of the routing protocol. Scalable Source Routing (SSR) proposed by Fuhrmann [13] introduces an MP2P architecture based on Chord. Similar to Chord, SSR is based on a ring architecture. However, SSR simplifies routing maintenance as no finger tables are used but only direct virtual neighbors. Further, physical neighbors are listed in the routing table. During routing, a node decides whether a virtual or physical neighbor is closer to the destination. Due to the simplified routing table, maintenance overhead is reduced. Further, by combining MANET and P2P routing tables, the route length can be reduced without increasing the complexity or the routing maintenance required.

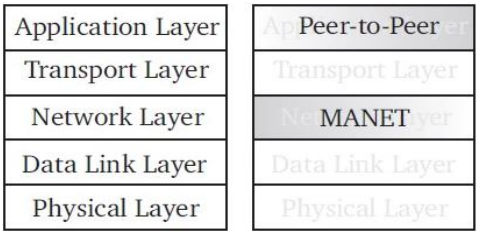
Additionally, awareness of physical neighbors at intermediate nodes between source and destination in the underlay can improve the efficiency of the routing protocol. The Virtual Ring Routing (VRR) algorithm presented in combines the set of virtual and physical neighbors in a single routing table at the network layer. As VRR is based on Ring architecture, the set of virtual neighbors consists of r nodes with a node-ID next to the node-ID of the routing table owner. Those r nodes consist of the r=2 virtual neighbors clockwise and the r=2 virtual neighbors counter-clockwise. VRR provides bidirectional links to the neighbors and maintains them proactively. The physical node set consists of the nodes in direct transmission range. The routing table provides the next underlay hop with an endpoint closest to the requested ID when an object is requested. As each intermediate node checks its own routing table before forwarding the request, requests may be rerouted when intermediate nodes are aware of shorter routes. This way, routing is simplified and overhead is reduced.

As a further optimization, every overheard or forwarded packet may be harnessed to update the routing tables in order to reduce maintenance overhead. This is the basic concept of EKTA and DPSR. Both integrated approaches implement a Pastry DHT on the network layer. Similar to Pastry, a primary and a leaf set routing table are provided. Complete multi-hop routes are stored at these routing tables to avoid underlay flooding during overlay routing. Shortest routes are preferred to reduce overhead. Whenever a message is overheard, the routing table is updated by routing information obtained from the message. Due to this, overhead generated by routing table maintenance is strongly reduced.

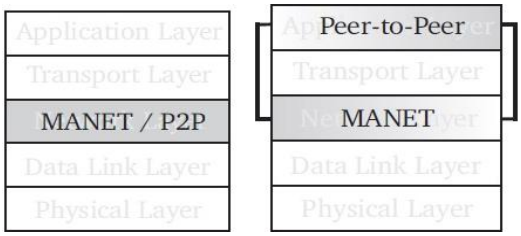
Cross Layered

Besides layered and integrated approaches, cross-layered architectures were proposed (as shown in Figure 4(d)). By providing network layer information to the P2P overlay, routing maintenance overhead can be reduced and the performance of the MP2P network can be improved.

CROSS Road combines a proactive MANET routing like the OLSR algorithm with a DHT overlay. As proactive routing algorithms have routes to all nodes of the network readily available, a cross layer approach could provide a complete knowledge of the network topology for the DHT. This way, each lookup may be accomplished after a single overlay hop. Yet, on the other hand, using a proactive routing protocol in a highly mobile, large scale network would result in a very high overhead. Similar to the layered approach presented in, MAD Pastry builds upon a landmark-based geographical node-ID distribution. In order to reduce the maintenance overhead, MAD Pastry truncates.



a) Basic Layered **b) Layered MP2P**



c) Integrated MP2P **d) Cross Layered MP2P**

Figure 4: MP2P Architectures

CHALLENGES

Peer-to-Peer

DHTs operate completely *distributed* and, due to this, no *trusted* entity for key management is available. Thus, offering security services such as access control in a *decentralized* P2P network is challenging. As outlined in the previous section, DHT lookup algorithms have to rely on the cooperation of other nodes in the network. However, this cannot be assumed in real-world applications and malicious nodes may misuse this weakness to launch attacks on the P2P network. As nodes may join and leave the network at any time, routing tables have to be *maintained* on a regular basis. Objects maintained by nodes which leave the network have to be managed by another node with an appropriate node-ID. This also may be misused by malicious nodes, since the information exchanged during the maintenance phase may be manipulated such that invalid entries are added to the routing table.

MANETs

MANETs are *distributed* and *decentralized* networks. No central entity is available to provide security services. Thus, security mechanisms for key distribution, access control, *etc.*, have to be implemented in a decentralized and, therefore, more complex way. Examples are approaches utilizing threshold cryptography to replace a central instance by a majority vote of the network nodes. Further, the available bandwidth is strongly limited and has to be shared with all nodes in transmission range. Due to this, countermeasures have to be limited in bandwidth requirements. In order to transmit messages to nodes not adjacent to the source, multi-hop transmissions are required.

Therefore, every node forwarding those messages has to be *trustworthy*. Otherwise, intermediate nodes showing a malicious behavior are able to redirect or drop messages instead of forwarding them correctly. As the nodes are *mobile*, the topology of a MANET changes permanently. Due to this, routes have to be adapted frequently and neighbors must be detected periodically. As mobile nodes are mostly battery-powered, energy is strongly limited. As adopted CPUs with low energy consumption are mostly deployed in mobile nodes, processing power is also limited. Due to this, complex and highly resource consuming cryptography might not be applicable for all scenarios. Further, as MANETs are *wireless* networks, they are susceptible to passive attacks as eavesdropping and, as the devices are mobile, they may be stolen and compromised.

Mobile Peer-to-Peer

As MP2P systems combine MANETs and P2P networks, challenges of both architectures are inherited. For instance, as routing table maintenance and object relocation of P2P networks requires a high amount of bandwidth and MANETs provide only *limited resources*, security algorithms have to be strongly limited in bandwidth. Further, as both architectures are *decentralized*, the availability of a Certification Authority (CA) cannot be assumed. Due to this, *trust* is a major problem as the routing algorithm has to rely on the cooperation of the nodes. As both architectures have a constantly *changing topology*, in the case of MANETs due to node mobility and in the case of P2P systems due to nodes going

offline, an increased amount of churn is generated compared to traditional P2P networks. This results in an increased overhead for object relocation.

ATTACKS

Attacking P2P

DHTs provide an efficient and decentralized lookup service. Yet, due to decentralization and multi-hop transmission, DHTs are vulnerable to attacks that exploit the lookup algorithm or the lack of admission control to deny services. We present selected attacks on DHTs in this chapter and discuss countermeasures that were proposed in the recent years.

Attacks on the P2P Routing Algorithm

The DHT lookup algorithm has to rely on the benign, cooperative behavior of the nodes that form the P2P network. Due to this, nodes that manipulate the lookup process have a serious influence on the network. During a Storage and Retrieval Attack [16], a malicious nodes denies the access to objects maintained by itself. As it is hard to distinguish between unavailable and denied objects during a lookup, most P2P systems are not inherently able to detect this attack. The efficiency of this attack depends directly on the fraction of malicious nodes, since objects are equally distributed to nodes.

Sit and Morris [16] introduced the Incorrect Lookup Routing attack. Malicious nodes drop or redirect incoming route requests instead of forwarding them to the correct destination. This is a simple and straight forward attack that affects the routing of lookup requests in DHTs. During the lookup process, a request is sent from the source to a node that is closer to the destination regarding the node-ID. Whenever a node receives a request, it checks whether it is the destination or whether it has to forward the request to the next overlay hop closer to the destination. For this recursive routing, Castro et al. introduced an analytical model that describes the effects of malicious nodes that drop packets (as shown in Equation 1).

The packet delivery ratio (σ) is based on the average number of hops (h) per request and the fraction of malicious nodes (f). While dropping lookup requests affects the routing process strongly, redirecting requests is a more subtle attack and may affect even DHTs protected by countermeasures.

$$\sigma = (1 - f)\, h$$

Attacks on the P2P Routing Tables

P2P routing tables have to be updated regularly to provide reliable routes when lookup requests have to be forwarded. Most DHTs update and optimize their routing tables by using entries provided by the routing tables of other nodes in the DHT. Malicious nodes may forge routing tables and provide routing table entries with links to inappropriate or non-existing nodes [16]. Further, corrupted routing updates can increase the fraction of routing table entries that refer to malicious nodes.

As most DHTs use metrics like proximity or age to optimize their routing table entries, malicious nodes may exploit these metrics to maximize the impact of the attack. Manipulation of the routing updates in order to increase the number of malicious nodes in a routing table of a benign node also results in a cascading effect. As each malicious node that is added to a routing table will provide routing updates, the fraction of malicious nodes increases as shown in Equation 2. The fraction of malicious routing entries *ftable* increases whenever a routing update is initiated as each benign node provides entries with a fraction of *fnetwork* malicious nodes, but each malicious node provides malicious entries only.

$$ftable = (1 - ftable) * fnetwork + ftable * 1$$

Sybil Attacks on P2P Networks

A malicious node may join a distributed network multiple times to generate multiple identities. A malicious node having multiple virtual identities can increase the impact of other attacks like the Incorrect Lookup Routing. Further, as P2P networks often use reputation mechanisms to avoid free riders, single nodes with multiple virtual identities may improve their own reputation.

Attacking MANETs

MANETs are vulnerable to several attacks due to their decentralized and wireless characteristics. In the following, we present selected attacks on MANETs that affect both the routing algorithm and the data transmission. We further discuss Sybil attacks in the context of MANETs.

Malicious Behaviour during Route Discovery and Maintenance

In order to transmit packets to nodes which are not in direct transmission range of the source node, intermediate nodes have to operate as routers to forward the packets to their destination. For this, the MANET routing algorithm has to rely on the cooperation of the intermediate nodes. Yet, the required cooperation cannot be assumed without restrictions in real-world environments. Misbehaving nodes may tamper with the routing algorithm by, e.g., forging, redirecting, or dropping routing messages. A wide range of routing attacks was introduced in the recent years. The Black hole attack is one of the most aggressive routing attacks on MANETs. Black holes manipulate the routing algorithm to redirect network traffic. For this, the routing metric is misused to increase the probability that the traffic is routed via the Black hole. As a result, the malicious node is capable of analyzing the transmitted data or of denying the service by dropping the received packets. Further, the routing algorithm may be misused for Resource Consumption attacks. We consider for example a reactive routing protocol and a malicious node that initiates a high number of route requests. As the bandwidth of MANETs is strongly limited and request messages are forwarded by each neighbor of the malicious node, those attacks can affect large parts of the network. Further, by forging the source field of the request messages, benign nodes can be accused for misbehavior and countermeasures can be evaded.

Besides an intended misbehavior, a selfish behavior can also affect the functionality of a MANET. Selfish nodes refuse to cooperate in acting as routers for other nodes in order to save resources. Since the functionality of MANETs is based on the cooperation of nodes, selfish nodes affect the availability of the network.

Malicious Behaviour during Data Transmission

After routes are established successfully, data transmission may still fail as a result of malicious behavior. Malicious nodes may e.g. cooperate during route discovery but deny to forward packets whenever data has to be transmitted. Those Selective Forwarding attacks may be hard to detect, especially when a node behaves benignly temporarily and only drops packets sent by specific nodes.

Further, by sending forged messages for route maintenance, transmission loops can be created.

Those loops result in an increased traffic and a denial of service as the packets are dropped when their time-to-live exceeds. The ADOV routing protocol for example has been shown to be vulnerable to Loop Forming attacks [17].

Sybil Attacks on MANETs

The susceptibility of MANETs to Sybil attacks is comparable to the susceptibility of P2P systems. MANETs are distributed networks and, therefore, in most scenarios no centralized admission control exists. Due to this, a single node

can generate multiple virtual identities in a network. Like in P2P systems, malicious nodes may generate multiple virtual identities to increase the impact of other attacks, to evade trust-based countermeasures, or to claim an increased amount of resources.

Attacking MP2P

MP2P networks are affected by both MANET and P2P attacks. Due to the characteristics of MP2P networks, (some of the) attacks may show more severe effects on MP2P networks than on MANETs or P2P networks individually.

As both systems are susceptible to insider attacks that deny cooperation during data transmission, MP2P networks are affected strongly by this kind of attacks. Especially in layered MP2P networks, denying correct forwarding on network layer would affect the packet delivery ratio and, thus, the lookup process strongly as, due to the multi hop data transmission on application and network layer, a high amount of network layer requests is required to perform a lookup for an object. Due to the high number of required hops, few malicious nodes in the network are sufficient to affect the routing process with a high probability [2].

Regarding Sybil attacks, MP2P networks may be attacked on both layers. Due to this, a malicious node will most probably attack the layer with the weakest countermeasure. Therefore, the probability of success for generating multiple identities in an MP2P network is increased compared to a MANET with a more robust overlay or a P2P system with a more robust underlay, respectively.

Further, MANET attacks can be modified to attack the P2P services. E.g., modified Black holes may redirect only routes to specific nodes to deny objects managed by them. On the other hand, malicious nodes may misuse P2P mechanisms like the routing table maintenance to increase the traffic and initiate a resource consumption attack on the MANET. Therefore, MANET robustness is required to provide reliable services by the P2P network but a benign behavior on the P2P network is also required as, otherwise, the underlying MANET can be affected.

CONCLUSIONS

In this paper, we discussed different approaches of combining MANETs and P2P systems into MP2P systems. We analyzed major challenges in securing MP2P networks by analyzing attack vectors of MANETs and P2P systems. We considered attacks such as Sybil or routing attacks that affect both MANETs and P2P systems and, thus, will also affect MP2P networks. We also concluded that the challenges in all three are somewhere related with each other especially with MP2P. As the attacks on these networks are more so there is a good space to further research on it.

REFERNCES

1. Schollmeier R., A Definition of *Peer-to-Peer* Networking for the Classification of *Peer-to-Peer* Architectures and Applications IEEE, 80333 Miinchen, Germany, 2002.
2. Gottron C, K.onig A and Steinmetz R, A Survey on Security in Mobile Peer-to-Peer Architectures - Overlay-Based vs. Underlay-Based Approaches, future internet 2010, ISSN 1999-5903
3. Christian Gottron, Andre K´ onig and Ralf Steinmetz, A Survey on Security in Mobile Peer-to-Peer Architectures, Overlay-Based vs. Underlay-Based Approaches, Future Internet 2010.
4. Steinmetz, R.; Wehrle, K.; G¨otz, S.; Schollmeier, R.; Rieche, S.; Ebersp¨acher, J.; Heckmann, O.; Darlagiannis, V.; Mauthe, A.; Koppen, C. Peer-to-Peer Systems and Applications; Springer: Berlin, Germany, 2005.

5. Kanchanasut, K.; Tunpan, A.; Awal, M.A.; Das, D.K.; Wongsaardsakul, T.; Tsuchimoto Y, A Multimedia Communication System for Collaborative Emergency Response Operation in Disaster-affected Areas; TR 2007-1; Technical Report, Internet Education and Research Laboratory (intERLab), Asian Institute of Technology (AIT): Bangkok, Thailand, 2007.
6. Industrieanlagen-Betriebsgesellschaft mbH. IABG–Info Com–HiMoNN—An efficient and highly mobile Ad-hoc Network Node, 2010.
7. Clausen, T.; Jacquet, P. Optimized Link State Routing Protocol (OLSR). RFC 2003, RFC 3626.
8. Perkins, C.; Belding-Royer, E.; Das, S. Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 2003, RFC 3561.
9. Castro, M.; Druschel, P.; Ganesh, A.; Rowstron, A.; Wallach, D. Secure routing for structured peer-to-peer overlay networks. In Proceedings of the 5th Symposium on Operating Systems Design and Implementation, Boston, MA, USA, 9–11 December 2002.
10. Harvesf, C.; Blough, D.M. The Effect of Replica Placement on Routing Robustness in Distributed Hash Tables. In Proceedings of the 6th IEEE International Conference on Peer-to-Peer Computing, Cambridge, UK, 6–8 September 2006.
11. Gopalan, A.; Znati, T. Peer Net: A peer-to-peer framework for service and application deployment in MANETs. In Proceedings of the 1st International Symposium on Wireless Pervasive Computing, Phuket, Thailand, 16–18 January 2006.
12. Douceur, J. The Sybil Attack. In Proceedings of the 1st International Workshop on Peer-to-Peer Systems, Cambridge, MA, USA, 7–8 March 2002.
13. Fuhrmann, T. Performance of scalable source routing in hybrid MANETs. In Proceedings of the 4th Annual Conference on Wireless on Demand Network Systems and Services, Obergurgl, Austria, 24–26 January 2007.
14. Maria Papadopouli, Henning Schulzrinne, peer-to-peer computing for mobile networks, Springer ISBN-13: 978-0-387-24427-3
15. Gopalan, A.; Znati, T. PeerNet: A peer-to-peer framework for service and application deployment in MANETs. In Proceedings of the 1st International Symposium on Wireless Pervasive Computing, Phuket, Thailand, 16–18 January 2006.
16. Sit, E.; Morris, R. Security Considerations for Peer-to-Peer Distributed Hash Tables. In Proceedings of the 1st International Workshop on Peer-to-Peer Systems, Cambridge, MA, USA, 7–8 March 2002.
17. Ning, P.; Sun, K. How to misuse AODV: A case study of insider attacks against mobile ad-hoc routing protocols. Ad Hoc Netw. 2005, 3, 795–819.